



Entire Screen Builder

	Version 5.2.1	Introducing Entire Screen Builder
--	---------------	-----------------------------------

This document applies to Entire Screen Builder Version 5.2.1 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

© Copyright Software AG 1999-2003
All rights reserved.

The name Software AG and/or all Software AG product names are either trademarks or registered trademarks of Software AG. Other company and product names mentioned herein may be trademarks of their respective owners.

Table of Contents

Introducing Entire Screen Builder	1
Introducing Entire Screen Builder	1
What is Entire Screen Builder?	2
What is Entire Screen Builder?	2
SDK (Software Development Kit)	4
Server	4
System Management Hub	5
Server Management	5
Presentation Clients (Viewers)	6
Transformation Rules	8
Transformation Rules	8
Built-in Rules	9
Basic Rules	9
Extended Rules	10
Examples for Using Transformation Rules	11
Examples for Using Transformation Rules	11
Using only the Built-in Rules	12
Using Basic Rules	13
Using Extended Rules	15
Overview of Features	17
Overview of Features	17
User and Group Profiles	20
User and Group Profiles	20
Scripting, User Exits and APIs	21
Scripting, User Exits and APIs	21
Scripting	21
Server-side User Exits	21
Client-side User Exits (APIs)	22
Natural UNIX User Exits	22

Introducing Entire Screen Builder

This documentation is organized under the following headings:

- | | | |
|---|---|--|
| ● | What is Entire Screen Builder? | General architecture and an overview of the Entire Screen Builder components. |
| ● | Transformation Rules | General information on the different types of transformation rules that can be defined for the GUI Version. |
| ● | Examples for Using Transformation Rules | Examples for the different levels of transformation that can be achieved in the GUI Version. |
| ● | Overview of Features | A list of the main features of Entire Screen Builder. |
| ● | User and Group Profiles | General information on the different types of profiles that can be defined using the System Management Hub. |
| ● | Scripting, User Exits and APIs | General information on linking your own processing logic to the system and on simplifying or automating repetitive tasks for the user. |

What is Entire Screen Builder?

Entire Screen Builder provides a complete range of host terminal communications in a client-server environment.

The central server provides access to applications running on OS/390, VSE/ESA, AS/400, BS2000, UNIX and OpenVMS. Presentation clients (viewers) are available for different styles of presentation, ranging from traditional green screen terminal emulation to "GUIified" Windows dialogs.

The legacy applications are automatically integrated in an Intranet environment without any effort, by using a browser as front-end. HTTP tunneling also allows to use the viewers in the Internet.

The GUI Version provides an easy to implement face-lift for character-mode host applications. Using the transformation rules defined with the Entire Screen Builder SDK, it transforms your legacy applications running on OS/390, VSE/ESA, AS/400, BS2000, UNIX and OpenVMS to graphical user interfaces (GUIs). The following applies to mainframe applications: transformation rules can be defined for applications written in any language.

Important:

On UNIX and OpenVMS, the GUI Version can only be used with Natural applications. With the GUI Version, it is not possible to access other UNIX applications or the UNIX operating system.

Many older applications have character-based user interfaces. Reengineering these interfaces to make them more user-friendly would extend the life of the applications, but it can require significant effort. Developers must know how the old interfaces are implemented, and what constraints the new interface must respect to remain compatible with the rest of the application. Using the Entire Screen Builder SDK, you can make the level of effort acceptable.

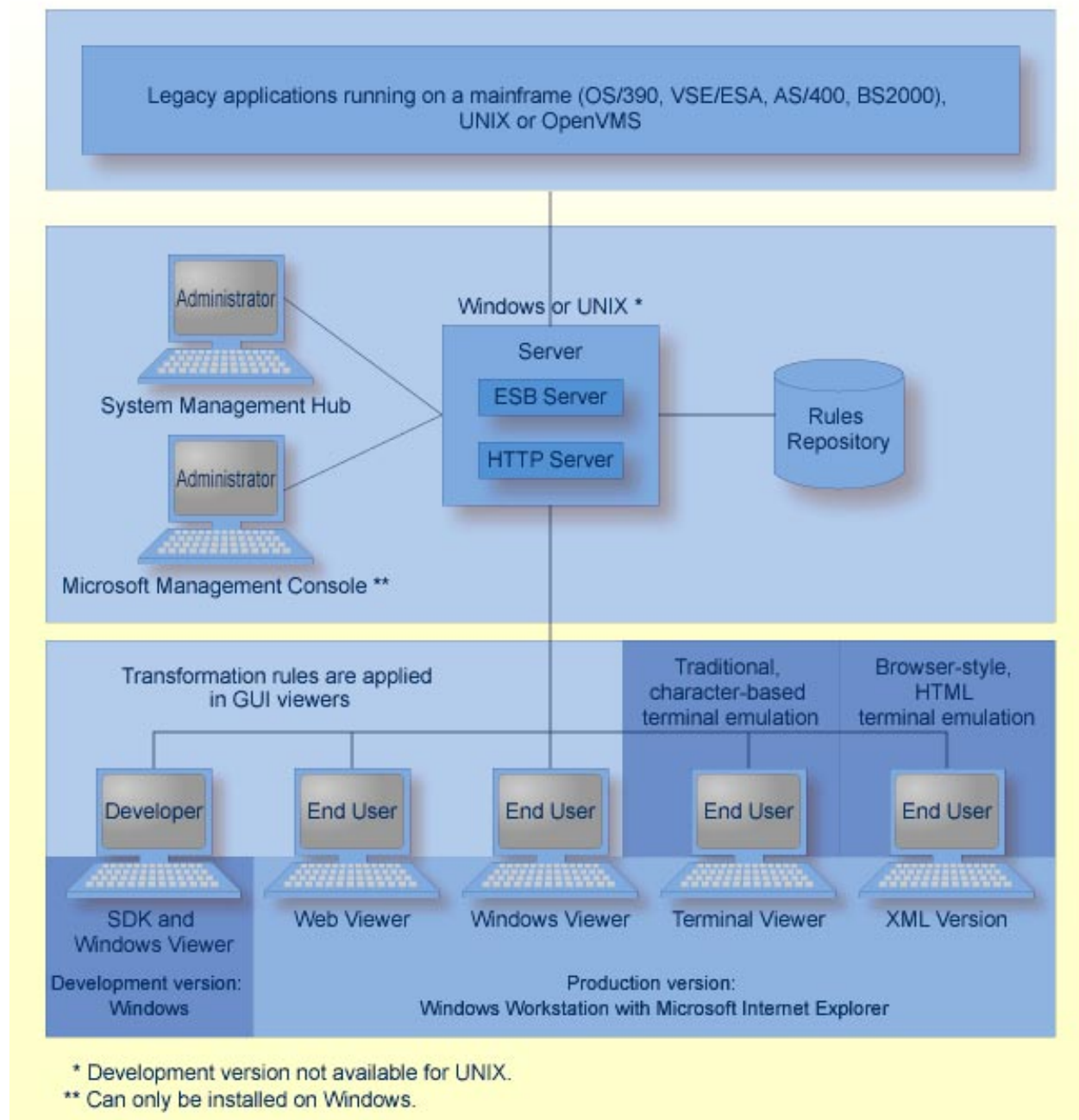
The Terminal Version provides traditional green screen terminal emulation.

The XML Version converts host screens to XML and HTML, allowing zero client footprint host access across a wide range of internet browsers.

Important:

On UNIX and OpenVMS, the XML Version can only be used with Natural applications. With the XML Version, it is not possible to access other UNIX applications or the UNIX operating system.

Entire Screen Builder's general architecture is depicted in the following graphic:



Entire Screen Builder consists of the following components:

- SDK (Software Development Kit)
 - Server
 - System Management Hub
 - Server Management
 - Presentation Clients (Viewers)
-

SDK (Software Development Kit)

This component is used by the developer to define the transformation rules for the GUI Version. These rules are later used by the server for transforming the character screens to Windows dialogs.

The rules are first created in the developer's local repository. The developer then has to copy the rules to the production environment to make them available to all users.

For detailed information, see the documentation *Defining the Rules Using the SDK*.

Server

The Entire Screen Builder Server handles both the communication with the host and with the viewers on the client workstations. Multiple and different types of viewers can connect to the server at the same time, each having one connection to a host system.

The Entire Screen Builder Server can be installed in one of the following environments:

- **Windows**
The Entire Screen Builder Server is automatically installed as a service on one of the supported Windows operating systems (see *Prerequisites* in the *Installation and Configuration* documentation). Configuration information is stored in the Windows registry and in the file *Nswconfig.xml*.
- **UNIX**
The Entire Screen Builder Server is installed as a normal UNIX process which does not require any special permissions or privileges. Configuration information is stored in the files *ewvreg.reg* and *nswconfig.xml*.

The Entire Screen Builder Server supports scripting for the host sessions. The script language which is used is the core JavaScript version 1.2 plus an Entire Screen Builder specific object which enables you to interact with the host session. See also: *Scripting, User Exits and APIs*.

System Management Hub

The administrator can use the System Management Hub (which is the standard, GUI-based, central point of administration for Software AG products) to administrate the Entire Screen Builder Server.

The System Management Hub is used to start and stop the Entire Screen Builder Server, to define and administrate host sessions as well as user and group profiles, to load and unload modules, to send messages to the clients currently connected to the server, etc.

For detailed information, see Entire Screen Builder's *System Management Hub* documentation.

Server Management

The Entire Screen Builder Server Management tool is a Microsoft Management Console snap-in which offers administration and monitoring facilities for the Entire Screen Builder Server. It allows the administrator to show the running connections and how the number of connections to the server varies with time. It can also be used to define and administrate host sessions, key schemes and color schemes. Host sessions can be added and newly added sessions can be changed without having to stop the Entire Screen Builder Server.

The Entire Screen Builder Server Management tool is intended to be used when the Entire Screen Builder Server is running, as a complement to the System Management Hub which has administration facilities but requires the Entire Screen Builder Server to be stopped for many functions.

For detailed information, see the *Server Management* documentation.

Presentation Clients (Viewers)

A presentation client is used by the end-user. Different presentation clients are available with the different Entire Screen Builder products.

GUI Version

The Entire Screen Builder GUI Version contains the GUI viewers in which the transformation rules are applied. The following viewers are available:

- Web Viewer running in a browser.
- Windows Viewer running as a separate Windows application.

Each character screen that the user invokes is transformed to a Windows dialog according to the transformation rules defined by the developer. Fields are thus converted to simple GUI controls such as edit boxes, option (radio) buttons or check boxes. The legacy applications work in the same way as before, without any modification or recompilation.

For further information, see the *GUI Viewers* documentation.

Terminal Version

The Entire Screen Builder Terminal Version contains the following viewers:

- Terminal Viewer running in a browser.
- Standalone Terminal Viewer running as a separate Windows application.

Transformation rules are not applied in the Terminal Viewer. It always shows the screen as sent from the host.

For further information, see the *Terminal Viewer* documentation.

XML Version

The Entire Screen Builder XML Version does not contain a special viewer. A standard internet browser (for example, Microsoft Internet Explorer 5 or 6) is used to access the host screens. The server output module of the XML Version converts the host screens to XML. In an HTTP server extension, the XML data stream is then translated to an HTML page using an XSL style sheet. Generic style sheets provided by Software AG or customer-supplied style sheets can be used.

The Entire Screen Builder Web Viewer and the Terminal Viewer are implemented as Microsoft ActiveX controls. Both have a COM interface that offers properties, methods and events which enable you to integrate the viewers into your Windows application and control the behavior of the ActiveX control. You can use the API from a wide variety of languages: C++, JavaScript, Visual Basic, etc. See also: *Scripting, User Exits and APIs*.

Terminal Model Support

The following table shows the terminal model support for the different presentation clients:

	Model 2 (24 x 80)	Model 3 (32 x 80)	Model 4 (43 x 80)	Model 5 (27 x 132)	UNIX
GUI Viewers	yes	no	no	no	24 x 80
Terminal Viewer	yes	yes	yes	yes	24 x 80
XML Version	yes	yes	yes	yes	24 x 80

Transformation Rules

Transformation rules are the key concept of the GUI Version. Once the customer application is started, the server converts every character-mode screen to a dialog according to the defined transformation rules. All rules, except the built-in rules, are defined by the developer.

If the application for which you plan to define transformation rules does not follow any standards, it will be a time-consuming task to define the rules. Therefore, before defining the transformation rules, it may be helpful to apply minor modifications to your legacy application in order to give it a homogeneous look. This step can include the use of similar application headers (titles, application names, dates), similar use of function keys and message lines, etc.

If your Natural application uses the technique of dynamically changing the fields of a map, you should turn the Natural output optimization off. Otherwise, Entire Screen Builder does not get consistent screen input. If Natural output optimization is turned on, the screens are sent in a compressed form. In this case, the field attributes may also be compressed. This makes recognizing and defining the rules for such a screen very complicated.

Use the Natural parameter `DCS=OFF` to turn the Natural output optimization off when you start Natural. You can also use the Natural statement `%ROOFF`. Note that `%ROOFF` only works correctly in the mainframe version of Natural 3.1 when ZAP NA46014 has been applied.

Entire Screen Builder distinguishes the following types of transformation rules, offering different levels of "GUIfication":

- Built-in Rules
- Basic Rules
- Extended Rules

The transformation rules are structured in scopes, where each scope can have its own set of transformation rules. A scope is entered when the screen detection defined for this scope is successful. If scope detection is not successful for a screen, global scope is used (if not defined otherwise with the **Last Rules Used** command). Scope detection is performed each time a new screen is received from the host. For further information, see *Using Different Scopes* in the documentation *Defining the Rules Using the SDK*.

Built-in Rules

The built-in rules cannot be changed by a developer. They guarantee that an application is executed by means of Entire Screen Builder as it is normally executed using the character interface.

The built-in rules are:

- Each output field in the character screen is converted to a static text control.
- Each input field in the character screen is converted to an edit box control.
- Double-clicking the left mouse button is equivalent to pressing the ENTER key.
- When the server application is running on a UNIX system, a special cursor is provided for fields with online help. Help can then be displayed by pressing the right mouse button.

Basic Rules

Basic rules immediately improve the user interface of an application with very little development effort.

Depending on the scope in which the basic rules have been defined, the rules are applied to all screens (global scope), only to screens of a specific application (application scope) or only to a specific screen (map scope).

Examples for basic rules are:

- Display the name of the application in the title bar of the application window.
- Display the fields sent by the application with a specific font or color.
- Provide command (push) buttons for all available host keys.
- Convert popup windows of the character screens to child windows.

For an overview of all basic rules, see the *Basic Rules Reference*.

Extended Rules

Extended rules provide the highest level of transformation. A character screen application can be transformed in such a way that it almost looks like a native Windows application. However, this requires quite some development effort.

If you want to create extended rules, you have to create a DLL using a resource editor such as Microsoft Visual Studio. This DLL must contain a dialog for each character screen on which the extended rules are to be applied. A dialog may contain controls such as option (radio) buttons, combo boxes and command (push) buttons. You then have to load the DLL into Entire Screen Builder's SDK in order to apply the extended rules. This includes the definition of the corresponding fields in the character screens.

Examples for extended rules are:

- Convert input fields consisting of several lines to edit boxes.
- Convert input fields in which one option from a number of mutually exclusive alternatives has to be entered to option (radio) buttons.
- Display list view controls for character screens with several rows where each row has an input field.
- Display tab controls for character screens that can be divided into several logical units.

For an overview of all extended rules, see the *Extended Rules Reference*.

Examples for Using Transformation Rules

The following examples are based on the same host screen. They show the different levels of transformation that can be achieved using Entire Screen Builder:

- Using only the Built-in Rules
 - Using Basic Rules
 - Using Extended Rules
-

Using only the Built-in Rules

When no rules have been defined, only the built-in rules are applied: the system function keys are provided, edit boxes are shown instead of input fields and double-clicking the left mouse button is equivalent to pressing the ENTER key. Not many differences can be noted in relation to the original character screen.

Entire Screen Builder Web Viewer					
05/06/00		* Viajes		14:31:57	
MCRUFECH		SAG TOURS		* Actividades	
=====\\		* Entretenimiento			
*** Consulta de Cruceros por Fecha ***					
F.Inicio	Pto. Salida	F.Fin	Pto. Destino	Precio	Plz Tipo
<input type="checkbox"/> 05/01/98	BAHAMAS	26/01/98	BAHAMAS	1,850,000	30 Medio
<input type="checkbox"/> 26/01/98	BAHAMAS	09/02/98	VIRGIN ISLANDS	125,555	13 Solo aloj.
<input type="checkbox"/> 09/02/98	VIRGIN ISLANDS	23/02/98	GUADELOUPE	2,500,000	14 Galera
<input type="checkbox"/> 02/03/98	GUADELOUPE	16/03/98	ST. LUCIA	1,200,000	13 Lujo
<input type="checkbox"/> 16/03/98	GUADELOUPE	30/03/98	ST. LUCIA	1,100,000	13 Lujo
<input type="checkbox"/> 30/03/98	ST. LUCIA	27/04/98	AZOREN	1,200,000	15 Solo aloj.
<input type="checkbox"/> 30/03/98	SAN GIORGIO D.N	05/04/98	DUBROVNIK	1,400,000	14 Galera
<input type="checkbox"/> 06/04/98	DUBROVNIK	19/04/98	KORFU	1,500,000	14 Lujo
<input type="checkbox"/> 20/04/98	KORFU	03/05/98	KEFALONIA	2,750,000	14 Lujo
<input type="checkbox"/> 20/04/98	ATHEN	03/05/98	KOS	1,555,555	14 Solo aloj.
F.Inicio desde: 01 01 98		<input checked="" type="checkbox"/> Lujo	<input checked="" type="checkbox"/> Solo alojamiento	Num. Cruc.: 216	
hasta: 05 06 00		<input checked="" type="checkbox"/> Medio	<input checked="" type="checkbox"/> Galera	Pagina....: 1	
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---					
Ayuda		Salir		Pag- MPral	
INS STAY					

Using Basic Rules

When basic rules have been defined, the above screen may be transformed as follows:

1 3 7 12

* Consulta de Cruceros por Fecha *

SOFTWARE AG

06/06/00
09:53:30

☒ Aceptar

Ayuda

Salir

Pag. -

M.Principal

	F.Inicio	Pto	Salida	F.Fin	Pto	Destino	Precio	Plz	Tipo
<input type="checkbox"/>	05/01/98		BAHAMAS	26/01/98		BAHAMAS	1,850,000	30	Medio
<input type="checkbox"/>	26/01/98		BAHAMAS	09/02/98		VIRGIN ISLANDS	125,555	13	Solo aloj.
<input type="checkbox"/>	09/02/98		VIRGIN ISLANDS	23/02/98		GUADELOUPE	2,500,000	14	Galera
<input type="checkbox"/>	02/03/98		GUADELOUPE	16/03/98	ST	LUCIA	1,200,000	13	Lujo
<input type="checkbox"/>	16/03/98		GUADELOUPE	30/03/98	ST	LUCIA	1,100,000	13	Lujo
<input type="checkbox"/>	30/03/98		ST LUCIA	27/04/98		AZOREN	1,200,000	15	Solo aloj.
<input type="checkbox"/>	30/03/98		SAN GIORGIO D.N	05/04/98		DUBROVNIK	1,400,000	14	Galera
<input type="checkbox"/>	06/04/98		DUBROVNIK	19/04/98		KOREU	1,500,000	14	Lujo
<input type="checkbox"/>	20/04/98		KOREU	03/05/98		KEFALONIA	2,750,000	14	Lujo
<input type="checkbox"/>	20/04/98		ATHEN	03/05/98		KOS	1,555,555	14	Solo aloj.

F.Inicio desde:
hasta:

☒ Lujo ☒ Solo alojamiento
☒ Medio ☒ Galera

Num Cruc 216
Pagina 1

MCRUFECH
INS STAY

Due to the defined basic rules, the following changes have been applied:


Rule Name	Description
Buttons for Main Dialog	Command (push) buttons are shown. They correspond to the host keys shown in the character screen. A line is shown between the buttons and the rest of the controls.
Delete Lines	The lines 1 to 5 of the original character screen have been removed.
Delete Prompt	The prompt characters "....:" have been removed.
Font and Colors	Another color (blue) has been defined for the text.
Frames	Two dialogs for which extended rules have been defined are used as frames. The top dialog contains two images, a black line and two static controls showing the current date and time. The bottom dialog contains an image and a black line.
Function Keys Toolbar	A function keys toolbar with default buttons is shown. The buttons correspond to the host keys shown in the character screen. Tooltips for function key buttons can be shown by moving the mouse over the toolbar buttons.
Item	The name of the screen has been moved to the title bar of the application window. The name of the map has been moved to the left side of the status bar.
Lines	The "----" lines in the character screen have been converted to Windows lines.

Using Extended Rules

When extended rules have been defined, the above screen may be transformed as follows:

1 3 7 12

Consulta de Cruceros por Fecha



Pagina 1

✓ Aceptar

?

Ayuda

Salir

← Pag -

→ Pag +

✗ Cancelar


F.Inicio	Pto.Salida	F.Fin	Pto.Destino	Precio	Plz	Tipo
05/01/98	BAHAMAS	26/01/98	BAHAMAS	1,850,000	30	Medio
26/01/98	BAHAMAS	09/02/98	VIRGIN ISLAN	125,555	13	Solo aloj.
09/02/98	VIRGIN ISLAN	23/02/98	GUADELOUPE	2,500,000	14	Galera
02/03/98	GUADELOUPE	16/03/98	ST. LUCIA	1,200,000	13	Lujo
16/03/98	GUADELOUPE	30/03/98	ST. LUCIA	1,100,000	13	Lujo
30/03/98	ST. LUCIA	27/04/98	AZOREN	1,200,000	15	Solo aloj.
30/03/98	SAN GIORGIO	05/04/98	DUBROVNIK	1,400,000	14	Galera
06/04/98	DUBROVNIK	19/04/98	KORFU	1,500,000	14	Lujo
20/04/98	KORFU	03/05/98	KEFALONIA	2,750,000	14	Lujo
20/04/98	ATHEN	03/05/98	KOS	1,555,555	14	Solo aloj.

Marcar el tipo de Crucero que se desea

☒ Lujo
☒ Medio
☒ Galera
☒ Solo Alojamiento

Seleccionar las fechas para realizar el Crucero

Desde 01 01 98 Hasta 06 06 00


216

MCRUFECH
INS STAY

The location of the controls in the dialog does not match the location of the fields in the character screen. Due to the defined extended rules, the following changes have been applied:

Rule Name	Description
Check Boxes	Check boxes are used for the input fields which are used to switch an option on or off.
Group Boxes	Group boxes are shown around elements that logically belong together.
List View Controls	A list view control is shown for the rows which have an input field in the character screen. Instead of entering an action code in an input field, the user has to press the right mouse button in order to select a command from a context menu.
Push Buttons	Command (push) buttons are shown on the left side of the dialog. They correspond to the host keys shown in the character screen. An image which does not invoke an action is shown at the top of the dialog. An image which invokes help is shown at the bottom right of the dialog. An image is a command (push) button for which the "Owner draw" style has been defined.
Static Text	Additional text which is not contained in the character screen is shown in the dialog.

In addition to the extended rules, the following basic rules have been applied:

Rule Name	Description
Font and Colors	Another font (MS Sans Serif) has been defined for the text.
Function Keys Toolbar	A function keys toolbar with default buttons is shown. The buttons correspond to the host keys shown in the character screen. Tooltips for function key buttons can be shown by moving the mouse over the toolbar buttons.
Item	The name of the screen has been moved to the title bar of the application window. The name of the map has been moved to the left side of the status bar.

Overview of Features

The main features of Entire Screen Builder are:

Zero mainframe footprint

The transformation rules and dialogs are defined and created by the developer without programming. This means, your legacy applications will continue to work in the same way, because no changes and no software are required on the server side.

Direct integration in a Web environment

The legacy applications are automatically integrated in an Intranet environment, without any effort. HTTP tunneling allows to use the viewers in the Internet. Tunneling is supported for the Microsoft Internet Information Server and the Apache Web Server. In combination with tunneling, the Web Viewer also supports the SSL (Secure Socket Layer) protocol.

Different types of output

Using the same transformation rules, the output can be shown in different GUI viewers: Web Viewer and Windows Viewer.

The Terminal Viewer offers traditional character-based terminal emulation with zero installation and administration on the client workstation.

The XML Version transforms the host screens to HTML which is sent to the browser on the client workstation. A standard internet browser is used to access the host screens.

Zero client installation and administration

The Web Viewer and the Terminal Viewer are automatically downloaded the first time they are used on a client machine or when the viewer version was updated.

Platform- and application-independent

Between 80 to 100 percent of the transformation rules and dialogs created for Natural applications running in a UNIX or OpenVMS system can be used for the same application running in a mainframe system (and vice versa). Entire Screen Builder handles and tolerates the most common platform differences. Fields are shifted one position to the left when going from mainframes to UNIX or OpenVMS. When the defined position of a field cannot be found, the following detection logic is used for the different platforms:

- For a mainframe session (3270 or BS2000), the server tries to find the field in the next position (+1).
- For a Natural UNIX or Natural OpenVMS session, the server tries to find the field in the previous position (-1).

Build-in multi-platform host communication

This is responsible for establishing the connection with the server side. For mainframe platforms, communication is based on the terminal protocol. For UNIX and OpenVMS platforms, it is based on TCP/IP and a direct communication protocol with Natural.

Block mode emulation

For UNIX and OpenVMS servers, and due to the client/server architecture of the product, Entire Screen Builder provides a block mode emulation to work with Natural applications. This feature can be very useful when using slow networks, for example TCP/IP under X.25.

Script language

A script language based on the JavaScript language allows to interact with the terminal session and to automate or simplify repetitive tasks for the users.

Browser-based administration

Software AG's System Management Hub, which is a browser-based online tool, is used to configure and administrate the Entire Screen Builder Server.

User and group profiles

The Entire Screen Builder Server can be accessed by anonymous or named users. It is possible to create a general user and group profile for all anonymous users. Named users can have their own profiles (for example, a specific logon script).

Administration and Monitoring

Statistics of the communication activity, the number of connections, connection types, times, buffer sizes etc. can be displayed in the Entire Screen Builder Server Management tool, which is a Microsoft Management Console snap-in. It is also possible to administrate the Entire Screen Builder Server.

User exits

There are several sets of user exit functions: a set for the server, a set for the Web Viewer, a set for the Terminal Viewer and a set for the Natural UNIX components of Entire Screen Builder.

Data compression between Natural UNIX and Natural OpenVMS applications and the Entire Screen Builder Server

The buffers sent from/to the UNIX and OpenVMS side can be compressed.

Data compression between the Entire Screen Builder Server and the Entire Screen Builder viewers

The buffers sent from/to the Entire Screen Builder Server can be compressed.

Data transfer

Data transfer with host applications written in Natural or SAP R/2 allows to upload data from the client to the host application and to download data from the host application to the client. Several PC data formats are directly supported.

User and Group Profiles

The Entire Screen Builder Server can be accessed from the viewers by anonymous or named users. A named user has to specify the user name and, if required, a password when connecting to the server.

User and group profiles allow to specify and control what a user is allowed to do and to give the user a specific environment, for example, another key scheme or a personalized logon script. Each named user may have a specific profile and can be member of a group, thus inheriting the group profile.

The administrator can also create one general user profile for anonymous users and can also create a group profile for them.

For further information, see *User and Group Concept* in Entire Screen Builder's *System Management Hub* documentation.

Scripting, User Exits and APIs

Entire Screen Builder offers different ways of linking your own processing logic to the system and to simplify or automate repetitive tasks for the user:

- Scripting
 - Server-side User Exits
 - Client-side User Exits (APIs)
 - Natural UNIX User Exits
-

Scripting

The Entire Screen Builder scripting language, which is based on the JavaScript language, allows to interact with host sessions (in a similar way in which the user interacts with sessions). It provides methods for detecting the screen contents and for sending data and terminal emulation keys to a host session. The script can be started by the user in a viewer. It is executed on the Entire Screen Builder Server. In terms of functionality, scripting can be compared with the server-side user exits. It provides, however, a higher, more user-oriented level.

The scripting language is available for all viewers (GUI viewers and Terminal Viewer). To write a script file for Entire Screen Builder, you should be familiar with the JavaScript language.

For detailed information, see the *Script Files* documentation.

Server-side User Exits

The server-side user exit is a dynamic link library (DLL) with user-written functions. It is called when a new screen arrives from the host and before a screen is sent back to the host. This allows to influence the way a screen is displayed by the viewer and to process and modify the data entered by the user before they are sent to the host application.

The server-side user exit is only available for the GUI viewers. To write such a user exit, you should be familiar with C++ programming and have a basic understanding of terminal emulation.

For detailed information, see *Server Functions* in the *User Exits* documentation.

Client-side User Exits (APIs)

The client-side user exit is a COM interface to the Web Viewer and the Terminal Viewer. These two viewers are implemented as Microsoft ActiveX controls. The API properties, methods and events allow you to integrate the viewers into your Windows application and control the behavior of the ActiveX control. You can use the API from a wide variety of languages: C++, JavaScript, Visual Basic, etc.

These client-side user exits are only available for the Web Viewer and Terminal Viewer, and they are not identical. To write such a user exit, you should be familiar with the use of Microsoft ActiveX control and the language you plan to use.

For detailed information, see the following topics in the *User Exits* documentation:

- *Web Viewer Client Functions (API)*
- *Terminal Viewer Client Functions (API)*
- *Common Viewer Functions (API)*

Natural UNIX User Exits

The Natural UNIX user exits are external libraries built on the UNIX machine to be called by the different Entire Screen Builder components running on the UNIX machine. They are only called by the Entire Screen Builder components in Natural UNIX, this means: the Natural UNIX user exits are never called by the Entire Screen Builder Server.

The Natural UNIX user exits are available for all types of viewers (GUI viewers and Terminal Viewer).

For detailed information, see *Natural UNIX User Exits* in the *User Exits* documentation.